PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appellant:	Michael Freed; Elango Ganesan; Praveen Patnala	Confirmation No.	4141
Serial No.:	09/900,515		
Filed:	July 6, 2001	Customer No.:	28863
Examiner:	Aravind K. Moorthy		
Group Art Unit:	2131		
Docket No.:	1014-056US01/JNP-0251		
Title:	SECURE SOCKETS LAYER CUT THROUGH ARCHITECTURE		

APPEAL BRIEF

Board of Patent Appeals and Interferences
Commissioner for Patents
Alexandria, VA 22313-1450

Sir:

This is an appeal from the final Office Action mailed on August 7, 2006 finally rejecting claims 1-8, 11-35, and 37-53, and the Notice of Panel Decision from Pre-Appeal Brief Review mailed on December 7, 2006 affirming the rejection of those claims.

Please charge Deposit Account No. 50-1778 the amount of $500.00 to cover the required fee for filing this Brief. Appellants request the opportunity for a personal appearance before the Board of Appeals to argue the issues of this appeal. The fee for the personal appearance will be timely paid upon receipt of the Examiner's Answer.

TABLE OF CONTENTS

REAL PARTY OF INTEREST

The Real Party of Interest is Juniper Networks, Inc., of Sunnyvale, California.

RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences for the above-referenced patent application.

STATUS OF CLAIMS

Claims 1-8, 11-35 and 37-53 are pending and are the subject of this Appeal (Appendix 1, Claims).

Claims 1-8, 11, 45-47, 51 and 53 stand rejected under 35 U.S.C. 102(e) as being anticipated by Ellis (USPN 6,484,257).

Claims 12, 14 and 48 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Fujiyama et al (USPN 6,052,728).

Claims 13 and 15 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Fujiyama et al. and in further view of Bellaton et al. (USPN 6,473,425).

Claims 16, 17 and 19 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Gelman et al. (USPN 6,415,329).

Claims 18 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Gelman et al. and in further view of Holtey et al. (USPN 5,293,424).

Claims 20-22, 27, 29, 33-35, 38, 39, 41 and 52 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. (USPN 6,253,337).

Claims 23-25 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. and in further view of Cohen et al. (USPN 6,389,462).

Claims 26 and 28 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. and in further view of Bellaton et al.

Claims 30 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. and in further view of Holtey et al.

Claim 31 stands rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. and in further view of Boeuf (USPN 6,009,502).

Claim 32 stands rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. and in further view of Weinstein et al. (USPN 6,094,485).

Claim 37 stands rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. and further in view of Harper et al. (USPN 6,820,215).

Claim 40 stands rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. and further in view of Bellaton et al.

Claim 42 stands rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. and further in view of Holtey et al.

Claim 43 stands rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. and further in view of Boeuf et al.

Claim 44 stands rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. and further in view of Weinstein et al. (USPN 6,094,485).

Claim 49 stands rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Holtey et al.

Claim 50 stands rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Harper et al. (USPN 6,820,215).

STATUS OF AMENDMENTS

No amendments have been filed subsequent to the Final Rejection mailed August 7, 2006 from which this Appeal has been made.

SUMMARY OF THE CLAIMED SUBJECT MATTER

In this appeal, Appellant separately argues independent claims 1, 20, 33 and 45. Consequently, independent claims 1, 20, 33 and 45 are involved in the appeal and summarized below.

Independent claim 1

Claim 1 recites *a method for secure communications between a client and a server.* Figure 3 and pg. 9, ln. 23 – pg.10, ln. 27 of the present application describe a secure socket layer (SSL) acceleration device as an intermediate device located between a web client and a web server.

Claim 1 requires *managing a communications negotiation between the client and the server through an intermediate device that supports a direct mode and a proxy mode.* Pg. 10, ll. 5-6 states that the SSL acceleration device supports a number of operational modes of encryption and decryption, including a direct (or pass-though) mode and a full proxy mode.

Claim 1 requires *receiving encrypted data packets from the client with the intermediate device, and decrypting each encrypted data packet with the intermediate device.* On pg. 9, ll. 2-5, the specification describes the SSL acceleration device as useful for offloading encryption and decryption tasks from a client or a server. Pg. 9, ll. 8-11, states that the SSL acceleration device operates to intercept secure communications between, for example, a Web based Internet client such as a Web browser operating on a personal computer, and a Web server. Pg. 9, ll. 29 - pg. 10, ln. 2 states that the SSL accelerator of Figure 3 performs SSL encryption and decryption and outputs decrypted data to the server.

Claim 1 also requires *forwarding unencrypted data packets from the intermediate device to the server using a communication session negotiated by the client and the server when the intermediate device operates in direct mode.* Pg. 8, ll. 13-16 and Figure 5 illustrate a sequence of communications between a client, an SSL accelerator device implementing a direct mode and a Web server. Pg. 16, ln 14 – pg. 17, ln. 20, describes in detail step 270 of Figure 5 illustrating the intermediate device operating in "direct mode" as processing encrypted application data to decrypt the data and forward the unencrypted (clear) data to the server using the session negotiated by the client and the server. With respect to the "direct mode," pg. 11, ll. 22-pg. 12, ln. 3 makes clear that in "direct mode," packets from client to server are addressed from the client to the server and from server to client. Specifically, the SSL accelerator allows the client and server to negotiate the TCP/IP session. Pg. 11, ll. 25-29. The specification states that this mode of the SSL accelerator is referred to as the "direct, cut-through" mode, since the client and server "think" they are communicating directly with each other. Pg. 12, ll. 1-3.

Claim 1 also requires *forwarding unencrypted data packets from the intermediate device to the server using a communication session negotiated by the server and the intermediate device when the intermediate device operates in proxy mode*. Pg. 8, ll. 22-26 and Figure 7 illustrate a sequence of communications between a client, an SSL accelerator device implementing a full TCP/IP and SSL proxy mode, and a Web server. In reference to Figure 7, the specification states that the SSL acceleration device, at step 207, performs all functions performed by the server and set forth in steps 206, 208 and 216 in Figures 5 and 6. At pg. 21, ll. 12-16, the specification states that at step 236, the SSL acceleration device 250 will negotiate its own TCP/IP session with server 300 to forward decrypted information to the server 300 in the clear. Further, with respect to proxy mode, pg. 9, ll. 13-16 states that the SSL acceleration device acts as a complete proxy, substituting itself for the server and both the TCP/IP handshaking sequence and the SSL encryption and decryption sequence. Pg. 12, ll. 15-17 states that, in full a full proxy mode, the SSL device acts as a proxy for one or more servers and handles both the SSL and TCP communications for the server. Pg. 21, ll. 7-9 states that, with respect to full proxy mode, the SSL accelerator performs a full proxy for both the TCP/IP negotiation process as well as the SSL encryption process.

Claim 1 requires *receiving data packets from the server; encrypting the data packets from the server; and forwarding encrypted data packets to the client*. Pg. 17, ln. 29 – pg. 18, ln 7 describes the SSL accelerator device receiving a clear (unencrypted) packet from the server, encrypting the data and forwarding the encrypted data packet to the client for a direct mode. Pg. 21, ll. 19-21, refers to Figure 7, block 280C, which the SSL acceleration device receiving clear data from the server, encrypting the packets within an HTTPS TCP session, and forwarding the encrypted data to the client in a full proxy mode.

Independent claim 20

Independent claim 20 requires *a method for secure communications between a client and one of a plurality of servers performed on an intermediary device*. Figure 3 and pg. 9, ln. 23 – pg.10, ln. 27 of the present application describe a secure socket layer (SSL) acceleration device as an intermediate device located between a web client and a web server.

Claim 20 requires *establishing a communications session between the client and said one of said plurality of servers by receiving negotiation data from the client intended for the server and forwarding the negotiation data in modified form to the server, and receiving negotiation data from the server intended for the client and forwarding the negotiation data to the client to establish the client and the server as terminations for the communications session.* The specification states that the SSL accelerator allows the client and server to negotiate the TCP/IP session directly, making only minor changes to the TCP/IP headers passing through the accelerator device. Pg. 11, ll. 25-28. Original claims 2-4 recite receiving TCP session negotiation data from the client and modifying the negotiation data prior to forwarding the data to the client, including modifying a SYN request from the client to the server and a maximum segment size to alter the packet transmission parameters.

Claim 20 requires *establishing a secure communications session between the client and the intermediary device.* Pg. 9, ll. 13-16 states that the SSL acceleration device can act as a complete proxy, substituting itself for the server for both the TCP/IP handshaking sequence and the SSL encryption and decryption sequence. Pg. 21, ll. 7-9 states that, with respect to full proxy mode, the SSL accelerator performs a full proxy for both the TCP/IP negotiation process as well as the SSL encryption process.

Claim 20 requires *maintaining a database of the secure communications session including information on the session/packet associations.* Pg. 11, ll. 28-29 of the specification states that the SSL accelerator tracks session data in a data structure in memory to enable SSL session handling to occur. Pg. 15, ll. 28 – 30, states that the SSL accelerator will update the TCP/SSL database and associate the SSL sequence numbers with the TCP sequence numbers for the session.

Claim 20 requires *receiving encrypted application data from the client at the intermediary device by the secure communications session between the intermediary device and the client*, and *decrypting the application data*. On pg. 9, ll. 2-5, the specification describes the SSL acceleration device as useful for offloading encryption and decryption tasks from a client or a server. Pg. 9, ll. 8-11, states that the SSL acceleration device operates to intercept secure communications between, for example, a Web based Internet client such as a Web browser operating on a personal computer, and a Web server. Pg. 9, ll. 29 - pg. 10, ln. 2 states that the

SSL accelerator of Figure 3 performs SSL encryption and decryption and outputs decrypted data to the server.

Claim 20 requires *forwarding decrypted application data from the intermediary device to said one of said plurality of servers using the communications session established between the client and the server.* Pg. 11, ll. 22-pg. 12, ln. 3 states that in a "direct mode," packets from client to server are addressed from the client to the server and from server to client, with the intermediary, SSL device being transparent to both. The specification states that the SSL accelerator allows the client and server to negotiate the TCP/IP session directly, making only minor changes to the TCP/IP headers passing through the accelerator device, and tracking session data in a data structure in memory to enable SSL session handling to occur. Pg. 11, ll. 25-29. The specification states that this mode is referred to herein as the "direct, cut-through" mode, since the client and server "think" they are communicating directly with each other, and the SSL accelerator is essentially transparent. Pg. 16, ln 14 – pg. 17, ln. 20, describes in detail step 270 of Figure 5 illustrating the intermediate device operating in direct mode to process encrypted application data to decrypt the data and forward the unencrypted (clear) data to the server using the session negotiated by the client and the server.

Independent claim 33

Claim 33 requires *an acceleration apparatus coupled to a public network and a secure network, communicating with a client via the public network and communicating with one of a plurality of servers via the secure network.* Figure 3 and pg. 9, ln. 23 – pg.10, ln. 27 of the present application describe a secure socket layer (SSL) acceleration device as an intermediate device located between a web client and a web server.

Claim 33 requires *a network communications interface; at least one processor; programmable dynamic memory; a communications channel coupling the processor, memory and network communications interface.* Figure 3 and pp. 10-11 describe a secure socket layer (SSL) acceleration device. Pg. 10, ll. 16-20 describe the SSL acceleration device as having network interface hardware, random access memory and a microprocessor.

Claim 33 requires *a client/server open communications session manager*, *a client secure communication session manager*, and *a client/server secure communications session tracking* *database*. Original claims 33-35 recites an open communications session manager and a secure communications session manager for secure and open (unencrypted) communications. Claim 33 requires. Pg. 11, ll. 28-29 of the specification states that the SSL accelerator tracks session data in a data structure in memory to enable SSL session handling to occur. Pg. 15, ll. 28 – 30, states that the SSL accelerator will update the TCP/SSL database and associate the SSL sequence numbers with the TCP sequence numbers for the session.

Claim 33 requires *a data packet encryption and decryption engine*. On pg. 9, ll. 2-5, the specification describes the SSL acceleration device as performing encryption and decryption tasks from a client or a server.

Claim 33 requires that *the acceleration apparatus is adapted to operate in a direct mode and a proxy mode*. Pg. 10, ll. 5-6 states that the SSL acceleration device supports a number of operational modes of encryption and decryption, including a direct (or pass-though) mode and a full proxy mode.

Claim 33 requires *wherein in the direct mode the acceleration apparatus decrypts data packets received from the client and forwards the decrypted data packets to one of the servers using a communication session negotiated by the client and the server*. Pg. 8, ll. 13-16 and Figure 5 illustrate a sequence of communications between a client, an SSL accelerator device implementing a direct mode and a Web server. With respect to the "direct mode," pg. 11, ll. 22-pg. 12, ln. 3 states that in "direct mode," packets from client to server are addressed from the client to the server and from server to client, with the intermediary, SSL device being transparent to both. The specification states that the SSL accelerator allows the client and server to negotiate the TCP/IP session. Pg. 11, ll. 25-29. The specification states that this mode is referred to herein as the "direct, cut-through" mode, since the client and server "think" they are communicating directly with each other, and the SSL accelerator is essentially transparent. Pg. 16, ln 14 – pg. 17, ln. 20, describes in detail step 270 of Figure 5 illustrating the intermediate device operating in direct mode to process encrypted application data to decrypt the data and forward the unencrypted (clear) data to the server using the session negotiated by the client and the server.

Claim 33 requires *wherein in the proxy mode the acceleration apparatus responds to the client on behalf of the server and forwards the decrypted data packets to the server using a communication session negotiated by the acceleration device and the server*. Pg. 8, ll. 22-26 and

Figure 7 illustrate the sequence of communications between a client, an SSL accelerator device implementing a full TCP/IP and SSL proxy mode, and a Web server. With respect to proxy mode, pg. 9, ll. 13-16 states that the SSL acceleration device acts as a complete proxy, substituting itself for the server and both the TCP/IP handshaking sequence and the SSL encryption and decryption sequence. Pg. 12, ll. 15-17 states that, in full a full proxy mode, the SSL device acts as a proxy for one or more servers and handles both the SSL and TCP communications for the server. Pg. 21, ll. 7-9 states that, with respect to full proxy mode, the SSL accelerator performs a full proxy for both the TCP/IP negotiation process as well as the SSL encryption process.

Independent claim 45

Claim 45 requires *a secure sockets layer processing acceleration device*, *a communication engine establishing a secure communications session with a client device via an open network*, and *a server communication engine establishing an open communications session with a server via a secure network*. Figure 3 and pg. 9, ln. 23 – pg.10, ln. 27 of the present application describe a secure socket layer (SSL) acceleration device as an intermediate device located between a web client and a web server. Original claims 34 and 35 state that the SSL device includes a client open communications session manager and secure communication manager enables the apparatus as a TCP and SSL proxy for the server, and a communications session managers enable transparent secure and open communication between the client and the server.

Claim 45 requires *an encryption and decryption engine operable on encrypted data packets received via the open communications session and on clear data received via the open communications session*. On pg. 9, ll. 2-5, the specification describes the SSL acceleration device as useful for offloading encryption and decryption tasks from a client or a server. Pg. 9, ll. 8-11, states that the SSL acceleration device operates to intercept secure communications between, for example, a Web based Internet client such as a Web browser operating on a personal computer, and a Web server. Pg. 9, ll. 29 - pg. 10, ln. 2 states that the SSL accelerator of Figure 3 performs SSL encryption and decryption and outputs decrypted data to the server.

Claim 45 requires *wherein the communication engine supports: (1) a direct mode in which decrypted data packets are forwarded to the servers using a communication session negotiated by the client and the server, and (2) a proxy mode in which the acceleration device responds to the client on behalf of the server and forwards the decrypted data packets to the server using the open communications session established by the acceleration device and the server.*

Pg. 10, ll. 5-6 states that the SSL acceleration device supports a number of operational modes of encryption and decryption, including a direct (or pass-though) mode and a full proxy mode.

Pg. 8, ll. 13-16 and Figure 5 illustrate a sequence of communications between a client, an SSL accelerator device implementing a direct mode and a Web server. Pg. 16, ln 14 – pg. 17, ln. 20, describes in detail step 270 of Figure 5 illustrating the intermediate device operating in "direct mode" as processing encrypted application data to decrypt the data and forward the unencrypted (clear) data to the server using the session negotiated by the client and the server. With respect to the "direct mode," pg. 11, ll. 22-pg. 12, ln. 3 makes clear that in "direct mode," packets from client to server are addressed from the client to the server and from server to client. Specifically, the SSL accelerator allows the client and server to negotiate the TCP/IP session. Pg. 11, ll. 25-29. The specification states that this mode of the SSL accelerator is referred to as the "direct, cut-through" mode, since the client and server "think" they are communicating directly with each other. Pg. 12, ll. 1-3.

Pg. 8, ll. 22-26 and Figure 7 illustrate a sequence of communications between a client, an SSL accelerator device implementing a full TCP/IP and SSL proxy mode, and a Web server. With respect to proxy mode, pg. 9, ll. 13-16 states that the SSL acceleration device acts as a complete proxy, substituting itself for the server and both the TCP/IP handshaking sequence and the SSL encryption and decryption sequence. Pg. 12, ll. 15-17 states that, in full a full proxy mode, the SSL device acts as a proxy for one or more servers and handles both the SSL and TCP communications for the server. Pg. 21, ll. 7-9 states that, with respect to full proxy mode, the SSL accelerator performs a full proxy for both the TCP/IP negotiation process as well as the SSL encryption process.

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Appellant submits the following grounds of rejection to be reviewed on Appeal:

1. The first ground of rejection to be reviewed on appeal is the rejection of claims 1-8, 11, 45-47, 51 and 53 as anticipated under 35 U.S.C. § 102(e) by U.S. Patent No. 6,484,257 to Ellis.
2. The second ground of rejection to be reviewed on appeal is the rejection of claims 20-22, 27, 29, 33-35, 38, 39, 41 and 52 under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. (USPN 6,253,337).

ARGUMENT

The First Ground of Rejection to be Reviewed on Appeal

Claims 1-8, 11, 45-47, 51 and 53 stand rejected under 35 U.S.C. 102(e) as being anticipated by Ellis (USPN 6,484,257). Appellant separately argues independent claims 1 and 45 and dependent claims 3, 51 and 53.

Independent claim 1

Claim 1 stands rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent No. 6,484,257 to Ellis. Claim 1 recites a method that requires managing a communications negotiation between the client and the server through an intermediate device that supports both a direct mode and a proxy mode. Thus, claim 1 requires an intermediate device that supports both a direct mode and a proxy mode when managing communications negotiations between a client and a server.

Claim 1 requires decrypting encrypted data packets with the intermediate device. Claim 1 also requires forwarding unencrypted data packets from the intermediate device to the server using a communication session negotiated by the client and the server when the intermediate device operates in direct mode. In addition, claim 1 requires forwarding unencrypted data packets from the intermediate device to the server using a second communication session negotiated by the server and the intermediate device when the intermediate device operates in proxy mode.

In this manner, claim 1 makes a distinction as to how the unencrypted data packets are forwarded by the intermediate device based on whether the intermediate device is operating in direct mode or proxy mode. As set forth above, claim 1 requires that, in direct mode, the intermediate device forward the unencrypted data using a communication session negotiated by the client and the server. Conversely, in proxy mode, the intermediate device forwards the unencrypted data packets from the intermediate device to the server using a communication session negotiated by the server and the intermediate device.

Thus, the literal requirement of claim 1 require that, depending on the mode, the Appellants' claimed intermediate device decrypts data packets and either uses a communication session negotiated by a client and a server to forward the decrypted data to the server (direct mode) or uses a separate session negotiated by the intermediate device and the server (proxy mode).

To aid the Board's understanding of these elements, Appellant refers the Board to Figure 3, reproduced below:

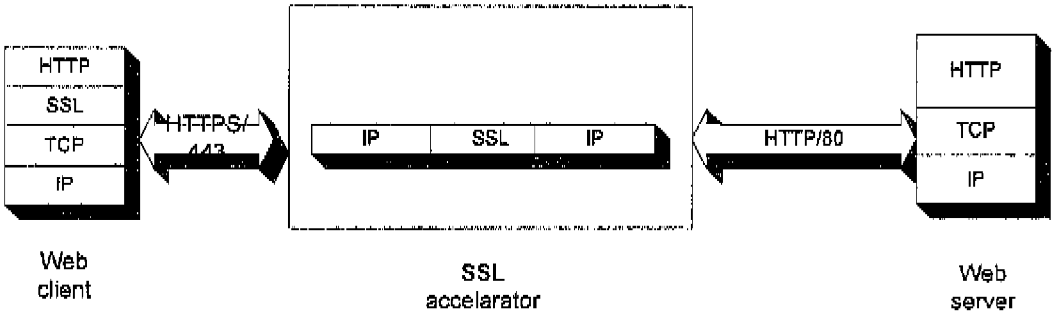

Figure 3 and pg. 9, ln. 23 – pg.10, ln. 27 of the present application describe a secure socket layer (SSL) acceleration device as an intermediate device located between a web client and a web server. Pg. 10, ll. 5-6 states that the SSL acceleration device supports a number of operational modes of encryption and decryption, including a direct (or pass-though) mode and a full proxy mode.

Operation of the SSL accelerator in "direct mode" is described in detail in reference to Figure 5 which shows that the SSL decrypts client communications and forwards the decrypted data using a communication session that was negotiated by the client and the server. In other words, the intermediate device provides SSL processing in a manner that is transparent to both the client and the server. The client and server negotiate a communication session with each other to form an end-to-end communication session that passes through the SSL acceleration device. The SSL accelerator intercepts encrypted client communications, provides SSL processing to decrypt the communications and injects the decrypted packets onto the same session for forwarding back to the server.

For example, as described in reference to step 270 of Figure 5, in direct mode the SSL accelerator receives encrypted data from the client, decrypts the data and forward the unencrypted (clear) packets to the server. Pg. 11, ll. 22-pg. 12, ln. 3 states that in "direct mode," packets from client to server are addressed from the client to the server and from server to client, with the intermediary, SSL device being transparent to both. The specification at pg. 11, ln. 25-27, makes clear that, in direct mode, the SSL accelerator allows the client and server to negotiate the TCP/IP session through the SSL accelerator, with the SSL accelerator making only minor changes to the TCP/IP headers passing through the accelerator device. Pg. 11, ln. 28-29 states that the SSL accelerator tracks session data in a data structure in memory to enable SSL session handling to occur. The specification at pg. 11, ln. 3- - pg. 12, ln 3 states that this mode is referred to as the direct mode because the client and server "think" they are communicating directly with each other, and the SSL accelerator is essentially transparent. Pg. 9, ll. 16-18 states that, in a one embodiment, the SSL acceleration device passes through the TCP/IP handshaking sequence and performs only SSL proxy encryption and decryption. In this manner, the SSL acceleration device provides SSL processing that is transparent to both the client and the server. That is, the SSL acceleration device decrypts data and, in a direct mode, forwards the decrypted data to the server using a communication session that was negotiated by the client and the server, as required by claim 1.

In contrast, in proxy mode, the SSL acceleration device negotiates separate communication sessions with the client and the server. For example, operation of the SSL device in proxy mode is described on pg. 8, ll. 22-26 in reference to Figure 7. With respect to proxy mode, pg. 9, ll. 13-16 states that the SSL acceleration device acts as a complete proxy, substituting itself for the server and both the TCP/IP handshaking sequence and the SSL encryption and decryption sequence. Pg. 12, ll. 15-17 states that, in full a full proxy mode, the SSL device acts as a proxy for one or more servers and handles both the SSL and TCP communications for the server. Pg. 21, ll. 7-9 states that, with respect to full proxy mode, the SSL accelerator performs a full proxy for both the TCP/IP negotiation process as well as the SSL encryption process. Thus, in full proxy mode, the SSL accelerator handles negotiation of the TCP/IP session.

Unlike direct mode, when in proxy mode the SSL accelerator forwards decrypted data to the server using a communication session negotiated by the intermediate device and the server, as required by claim 1. In reference to proxy mode, the specification at pg. 21, ll. 12-16, states that at step 236 of Figure 7, the SSL acceleration device 250 will negotiate its own TCP/IP session with server 300 to forward decrypted information to the server 300 in the clear, i.e., decrypted form. Further, with respect to proxy mode, pg. 9, ll. 13-16 states that the SSL acceleration device acts as a complete proxy, substituting itself for the server and both the TCP/IP handshaking sequence and the SSL encryption and decryption sequence. Pg. 12, ll. 15-17 states that, in full a full proxy mode, the SSL device acts as a proxy for one or more servers and handles both the SSL and TCP communications for the server. Pg. 21, ll. 7-9 states that, with respect to full proxy mode, the SSL accelerator performs a full proxy for both the TCP/IP negotiation process as well as the SSL encryption process. In this manner, the SSL acceleration device provides SSL processing to decrypt client communications and, in proxy mode, forwards decrypted data to the server using a communication session negotiated by the intermediate device and the server (not a session negotiated by the client and the server).

Ellis

Ellis describes a distributed architecture that provides a software solution to the major computational challenges faced with providing secure communication.[1] The Ellis system includes clients, a main server, and software agents executing on destination devices, also referred to as agent servers. This architecture is shown in Figure 2 of Ellis, reproduced below:

[1] Ellis at Abstract

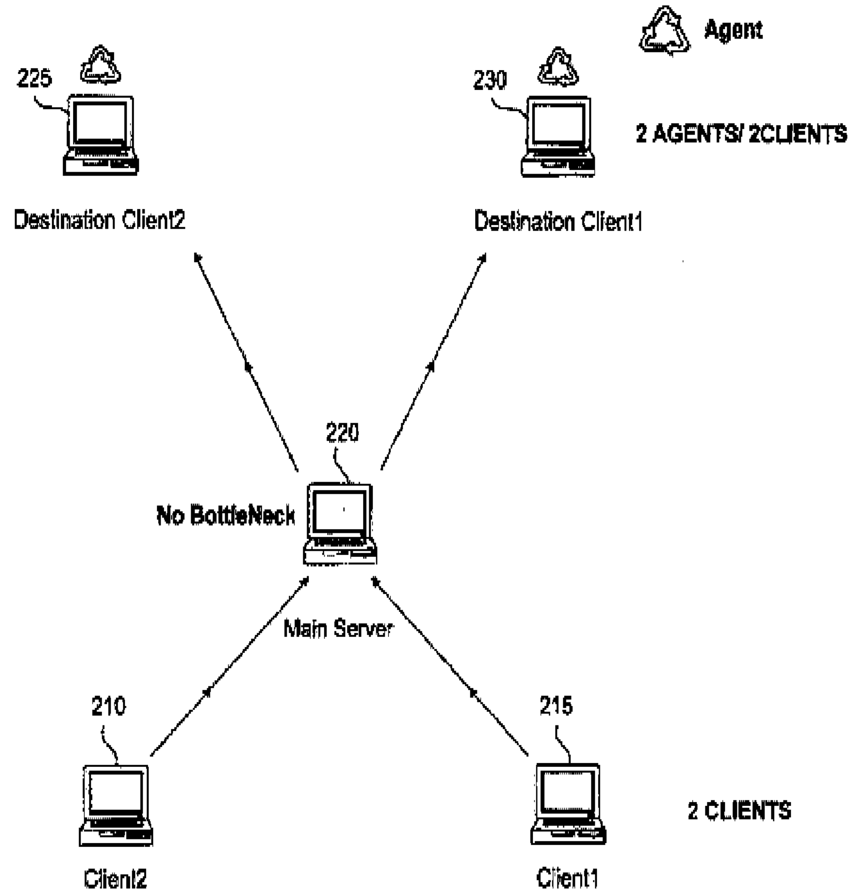

Fig. 2

Ellis explains that when the Main Server starts up, a registry is created and initialized and the Main Server begins execution.[2] The Agent Servers register themselves with the Main Server and define session keys with which to establish secure communications.[3] This allows both the Main Server and Agent Servers become enabled to receive and process secure connections from clients on behalf of final destinations.

When a client connects to the Main Server, the Main Server determines whether it has sufficient resources to service the secure session.[4] If so, the Main Server accepts the session and handles the secure communications.[5] If the Main Server has insufficient resources, then it

[2] *Id.* at col. 7, ll. 17-18.
[3] *Id.* at col. 7, ll. 19-22.
[4] *Id.* at col. 7, ll. 25-27.
[5] *Id.* at col. 7, ll. 28-29.

instructs an Agent Server to wake up and handle the session.[6] In this way, either the Main Server or the Agent Servers can act as intermediate devices for the final destinations. If none of the Agent Servers can handle the session, the Main Server can attempt to handle the session or deny the connection.[7]

Ellis makes clear that if an Agent Server is used to handle SSL processing, that Agent Server and the Client independently negotiate a communication session. Specifically, Ellis states that, in the event an Agent Server is used, the Main Server notifies both the Client and the Agent Server of each other's addresses, and the Client and the Agent Server independently generate a session key to exchange data.[8] The Agent Server is then responsible for decrypting the session communications from the client and directing the decrypted communication to the intended final destination.[9] In this manner, the Ellis System is capable of handing off the encryption / decryption functions to Agent Servers, where the Agent Servers independently negotiate sessions with the Clients and perform decryption services for the client communications.

In rejecting claim 1, the Examiner asserts that Ellis describes an intermediate device that supports both a direct mode and a proxy mode as defined by Appellants' claims.[10] Specifically, the Examiner clarified his position on page 2 of the Final Office Action as follows:

> *Ellis discloses a direct mode. The direct mode as taught by Ellis is when the clients are communicating directly without interference of the 'main server". The proxy mode is when communication goes through the main server.*

This statement reveals the Examiner's errors. First, as made clear by this statement, the Examiner asserts that the Ellis system includes an intermediate device (the "main server") that operates in a "direct mode" when the clients communicate directly "without interference" from the main server. The Examiner distinguishes this from proxy mode where "communication goes through the main server."

This, however, overlooks the fundamental requirements of Appellant's claim 1. For example, claim 1 specifically requires *forwarding unencrypted data packets from the intermediate device to the server using a communication session negotiated by the client and the*

[6] *Id.* at col. 7, ll. 30-32.
[7] *Id.* at col. 7, ll. 36-37.
[8] *Id.* at Col. 7, ll. 51-53.
[9] *Id.* at Col. 7, ll. 57-59.
[10] Final Office Action mailed 8/7/2006, pg. 2.

server when the intermediate device operates in direct mode. In other words, claim 1 requires that the intermediate device that decrypted the data packets forward unencrypted data packets to the server using a session that the client and server negotiated when operating in direct mode. In this manner, the direct mode allows the intermediate device to receive encrypted data packets, decrypt the data packets and then forward unencrypted data packets to the server using the session that the client and server negotiated. This language requires at least two elements not addressed by the Examiner's reasoning.

First, claim 1 requires that, in direct mode, the intermediate device forward unencrypted data. The Examiner's assertion that Ellis describes an intermediate device that supports a "direct mode" because the clients communicate directly "without interference" from the Main Server overlooks the claim requirements that Appellant's claimed intermediate device decrypts the data and, in direct mode, forwards the unencrypted data to the server. Ellis specifically states that if an Agent Server is used, that Agent Server decrypts the session communications from the client and directs the decrypted communication to the intended final destination. Col. 7, ll. 57-59. Therefore, when the clients communicate directly without interference from the Main Server, as asserted by the Examiner, the Main Server does not perform the function of decrypting the data packets. To the contrary, as the Examiner seems to recognize, when the client communicates with the Agent Server, the Agent server decrypts the data packets and the Main Server is effectively bypassed. Thus, the Main Server is not forwarding unencrypted data packets in a "direct mode" at all.

Second, claim 1 requires that, in direct mode, the intermediate device forward <u>unencrypted</u> data <u>using a session negotiated by a client and the server</u>. This requires that the intermediate device forward the unencrypted data using a session negotiated by other devices and, in particular, a client and a server. Contrary to the Examiner's reasoning, when the a client communicate directly with the Agent Server, the Main Server does not, when operating in direct mode, forward <u>unencrypted</u> data packet to a server support <u>using a communication session negotiated by the client and the server</u>. The Examiner elaborated on this point at page 3 of the Final Office Action by stating:

> *Appellant argues that [Ellis fails to teach that] the intermediate device operates in a direct mode to decrypt data encrypted data packets and forward unencrypted data*

> *packets from the intermediate device to the server using a communication session negotiated by the client and the server.*
>
> *The examiner disagrees. Ellis discloses forming a session between a client and the agent server. The agent server decrypts the session communication and redirects the decrypted data to its final destination.*

Therefore, the Examiner's position is that when a Client communicates directly with an Agent Server, the Main Server supports a direct mode by forward unencrypted data packets using a session negotiated by those devices.

However, as recognized by the Examiner in the above quoted statement, Ellis makes clear that, when an Agent Server is used, the Agent Server (not the Main Server) decrypts the session communication and redirects the decrypted data to its final destination. Consequently, the Main Server in Ellis does not qualify as an intermediate device that supports a direct mode as described and claimed by the Appellant. When the Agent Server handles a client connection, the Agent Server negotiations the session, decrypts the data and directs the data to the final destination. The Main Server is not involved and certainly does not forward unencrypted data using a communication session negotiated by the client and the server.

In summary, neither the Agent Server nor the Main Server in Ellis teach or suggest a direct mode in which an intermediate device utilizes a session that it did not negotiate, i.e., a session that a client and a server negotiated, to forward decrypted data packets to that server, as required by claim 1. Quite the opposite, the Main Server and the Agent Servers in Ellis appear to be operating as the classical proxy servers for the sessions handled by these devices on behalf of final destinations. The Main Server and the Agent Servers independently negotiate communications sessions with the clients, and certainly do not use sessions negotiated by other devices (i.e., the client and the server) for forwarding decrypted data. When the Main Server cannot handle a client communication session, it hands off the communication session to an Agent Server and does not handle any SSL processing for that session. Thus, in either case, the device handling the SSL processing communicates with the client and the final destination using sessions that it negotiated.

For at least these reasons, Ellis fails to teach or suggest an intermediate device that supports both a direct mode and a proxy mode in the manner required in claim 1. The intermediate devices of Ellis (i.e., the Main Server or the Agent Servers) only operate as proxies that independently negotiate secure communications sessions with the clients and process secure communications. There is no teaching or suggestion in Ellis of decrypting encrypted data packets with an intermediate device, and forwarding unencrypted data packets from the intermediate device to the server using a communication session negotiated by the client and the server when the intermediate device operates in direct mode, as required by claim 1.

For at least these reasons, Ellis fails to anticipate the requirements of independent claim 1. Moreover, none of the other references, either singularly or in combination, provide any teaching or suggestion that overcomes the deficiencies of Ellis. The Board should reverse the rejection of claim 1 under 35 U.S.C. 102(e) as being anticipated by Ellis.

Independent claim 45

Claim 45 requires wherein the communication engine supports: (1) a direct mode in which decrypted data packets are forwarded to the servers using a communication session negotiated by the client and the server, and (2) a proxy mode in which the acceleration device responds to the client on behalf of the server and forwards the decrypted data packets to the server using the open communications session established by the acceleration device and the server.

As explained above, none of the intermediate devices of Ellis (i.e., the Main Server or the Agent Servers) has a communication engine that support two different modes for forwarding decrypted data to a server, as required by claim 45. Moreover, no device in the Ellis system includes a communication engine that supports a direct mode in which decrypted data packets are forwarded to the servers using a communication session negotiated by the client and the server. The Board should reverse the rejection of claim 45 under 35 U.S.C. 102(e) as being anticipated by Ellis.

Dependent claim 3

Appellant separately argues dependent claim 3, which recites modifying a SYN request from the client to the server to alter the packet transmission parameters.

With respect to dependent claim 3, nothing in Ellis suggests modifying a SYN request. In fact, in its entirety, Ellis does not even refer to a SYN request, let alone describe modification of a SYN request by an intermediate device.

On this point, the Examiner stated that in Ellis "the SYN request is modified by the decryption of the requests. The requests are being altered from an encrypted to a decrypted state."[11] Appellant submits that this is factually incorrect. SYN requests are part of a TCP/IP handshake used to establish TCP/IP sessions. Only after a TCP/IP connection is established does a key exchange occur and an SSL session is established. Encryption / decryption of data cannot occur until after the TCP/IP and SSL sessions are established, i.e., after the TCP handshake. In summary, Appellant finds no evidence in Ellis that SYN requests are "modified" from an encrypted form to a decrypted form, as suggested by the Examiner.

The Board should reverse the rejection of claim 3 under 35 U.S.C. 102(e) as being anticipated by Ellis.

Dependent claims 51 and 53

Appellant separately argues dependent claim 51 and 53 as a group. Claim 51 depends on claim 1 and requires automatically switching the intermediate device from the direct mode to the proxy mode upon detecting a communication error associated with the direct mode. Claim 53 depends on claim 45 and requires that the SSL acceleration device include a communications engine that automatically switches from the direct mode to the proxy mode upon detection of a communication error with the communication session negotiated by the client and the server.

In rejecting claims 51 and 53, the Examiner cited Ellis at col. 7, ln. 11 to col. 8, ln. 27 without providing any additional analysis. Appellant submits that Ellis does not describe an intermediate device that includes a communications engine that automatically switches from the direct mode to the proxy mode upon detection of a communication error with the communication session negotiated by the client and the server.

First, as discussed above, the Examiner erred in suggesting that the Main Server in the Ellis system supports a direct mode as required by the language of claims 1 or 45. Moreover, Ellis certainly fails to describe intermediate device that includes a communications engine that

[11] Final Office Action, pg. 3, ll. 11-13.

automatically switches from the direct mode to the proxy mode upon detection of a communication error with the communication session negotiated by the client and the server. As discussed above, the Examiner's position is that the direct mode is taught by Ellis when the clients are communicating directly without interference of the main server, and that the proxy mode is when communication goes through the main server.[12] This fails to teach or suggest a communication engine that automatically switches from a direct mode to the proxy mode upon detection of a communication error with the communication session negotiated by the client and the server.

The Second Ground of Rejection to Be Reviewed on Appeal

Claims 20-22, 27, 29, 33-35, 38, 39, 41 and 52 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Ellis in view of Maloney et al. (USPN 6,253,337). Appellant separately argues independent claims 20 and 33.

Independent claim 20

Claim 20 recites a method for secure communications between a client and one of a plurality of servers performed on an intermediary device. Claim 20 also requires establishing a communications session between the client and said one of said plurality of servers by receiving negotiation data from the client intended for the server and forwarding the negotiation data in modified form to the server, and receiving negotiation data from the server intended for the client and forwarding the negotiation data to the client to establish the client and the server as terminations for the communications session. The Examiner cites Ellis at col. 8, ln. 54 to col. 9, ln. 49 as teaching these elements and provides no additional analysis.[13]

Claim 20 further requires forwarding decrypted application data from the intermediary device to said one of said plurality of servers using the communications session established between the client and the server, as required by claim 20. With respect to these elements, the Examiner again relies on Ellis citing col. 8, ln. 54 to col. 9, ln. 49 with no additional analysis.[14]

As explained above, the intermediate devices of Ellis (i.e., the Main Server or the Agent Servers) independently negotiate secure communications sessions with the clients and process

[12] Final Office Action, pg. 2.
[13] Final Office Action, pg. 12.
[14] Final Office Action, pg. 12.

secure communications. No intermediate devices in Ellis decrypts data and, in a direct mode, forwards decrypted data packets from the intermediate device to the server using a communication session negotiated by the client and the server. In contrast, Ellis makes clear that the Main Server or the Agent Servers negotiate directly with the client and, therefore, simply do not utilize sessions negotiated by the client and server to forward decrypted data to the server.

Neither Maloney et al. nor any of the other references overcome the Examiner's error with respect to Ellis. The Examiner's only reliance on Maloney et al. is with respect to the claim element of maintaining a database of the secure communications session including information on the session/packet associations. Maloney et al. describes an information analysis system that is a combination of sensor, analysis, data conversion, and visualization programs.

The Board should reverse the rejection of claim 20 under 35 U.S.C. 103 as being unpatentable over Ellis in view of Maloney et al.

Independent claim 33

Claim 33 requires an acceleration apparatus adapted to operate in a direct mode and a proxy mode. Claim 33 requires that, when in the direct mode, the acceleration apparatus decrypts data packets received from the client and forwards the decrypted data packets to one of the servers using a communication session negotiated by the client and the server. Claim 33 also requires that, when in the proxy mode, the acceleration apparatus responds to the client on behalf of the server and forwards the decrypted data packets to the server using a communication session negotiated by the acceleration device and the server. With respect to these elements, the Examiner cites Ellis at col. 8, ln. 54 to col. 9, ln. 49 and provides no additional analysis.[15]

As explained above, none of the intermediate devices of Ellis (i.e., the Main Server or the Agent Servers) support two different modes for forwarding decrypted data to a server, as required by claim 33. No intermediate device in Ellis (i.e., the Main Server or the Agent Servers) supports such a mode. Ellis makes clear that the Main Server or the Agent Servers negotiate directly with the client and, therefore, simply do not utilize sessions negotiated by the client and server to forward decrypted data to the server.

[15] Final Office Action, pg. 21.

To be clear, claim 33 requires that, in direct mode, the acceleration device decrypts data packets and forwards decrypted data to the server using a communication negotiated by the client and the server. The Examiner's argument that the Main Server supports a direct mode by not interfering with the communications between the client and the Agent Servers overlooks that, in this situation, the Main Server is not performing the function of decrypting the secure data from the client nor forwarding decrypted data at all, let alone using a session negotiated by the client and the server. Claim 33 literally requires that, in direct mode, the acceleration device performs both functions of decrypting data packets and forwarding decrypted data to the server using a communication negotiated by the client and the server. These elements are not taught or suggested by Ellis in view of Maloney et al.

Neither Maloney et al. nor any of the other references overcome the Examiner's error with respect to Ellis. The Examiner's only reliance on Maloney et al. is with respect to the claim element of a client/server secure communications session tracking database. As explained above, Maloney et al. describes an information analysis system that is a combination of sensor, analysis, data conversion, and visualization programs.

The Board should reverse the rejection of claim 33 under 35 U.S.C. 103 as being unpatentable over Ellis in view of Maloney et al The Board should reverse the rejection of claim 33 under 35 U.S.C. 102(e) as being anticipated by Ellis.

It is earnestly requested that the Examiner's rejection be reversed, and that all of the pending claims be allowed.

Date:

February 7, 2007
SHUMAKER & SIEFFERT, P.A.
8425 Seasons Parkway, Suite 105
St. Paul, Minnesota 55125
Telephone: 651.735.1100
Facsimile: 651.735.1102

By:

Kent J. Sieffert
Name: Kent J. Sieffert
Reg. No.: 41,312

APPENDIX - CLAIMS ON APPEAL

Claim 1 (Previously Presented): A method for secure communications between a client and a server, comprising:

managing a communications negotiation between the client and the server through an intermediate device that supports a direct mode and a proxy mode;

receiving encrypted data packets from the client with the intermediate device;

decrypting each encrypted data packet with the intermediate device;

forwarding unencrypted data packets from the intermediate device to the server using a communication session negotiated by the client and the server when the intermediate device operates in direct mode;

forwarding unencrypted data packets from the intermediate device to the server using a communication session negotiated by the server and the intermediate device when the intermediate device operates in proxy mode;

receiving data packets from the server;

encrypting the data packets from the server; and

forwarding encrypted data packets to the client.

Claim 2 (Previously Presented): The method of claim 1 wherein said step of managing comprises:

receiving TCP session negotiation data from the client and modifying the negotiation data prior to forwarding the negotiation data to the server to establish the communications session between the client and the server when operating in direct mode.

Claim 3 (Original): The method of claim 2 wherein the method includes the further step of modifying a SYN request from the client to the server to alter the packet transmission parameters.

Claim 4 (Original): The method of claim 3 wherein said step of modifying includes modifying at least a maximum segment size value of said data packet.

Claim 5 (Previously Presented): The method of claim 1, wherein the method further includes the steps of negotiating an SSL session with the client.

Claim 6 (Previously Presented): The method of claim 1 wherein decrypting comprises decrypting SSL encrypted packet data, and wherein encrypting comprises encrypting a data packet with SSL.

Claim 7 (Previously Presented): The method of claim 1 wherein said step of managing comprises receiving with the intermediate device communication negotiation data directed to the server from the client and responding to said negotiation in place of the server when the intermediate device operates in proxy mode.

Claim 8 (Previously Presented): The method of claim 7 further including negotiating the communications session between the server and the intermediate device as a separate TCP session.

Claims 9-10 (Cancelled).

Claim 11 (Previously Presented): The method of claim 1 further including the step, prior to said step of receiving encrypted data, of negotiating an encrypted data communications session between the intermediate device and the client.

Claim 12 (Original): The method of claim 1 wherein said step of managing comprises maintaining a database of entries on each session of data packets communicated between the client and the server.

Claim 13 (Original): The method of claim 12 wherein said database includes an entry for a session comprising a session ID, a TCP Sequence number and an SSL session number.

Claim 14 (Original): The method of claim 12 wherein said entry further includes an initialization vector.

Claim 15 (Original): The method of claim 12 wherein said entry includes an expected ACK.

Claim 16 (Original): The method of claim1 wherein said step of receiving encrypted data packets includes receiving data packets including encrypted application data spanning multiple packets, and said step of forwarding includes forwarding a portion of the application data contained in an individual encrypted TCP segments to the server without authentication.

Claim 17 (Original): The method of claim 16 further including the step of authenticating the application data on receipt of all packets including the application data.

Claim 18 (Original): The method of claim 16 wherein said data is not buffered during decryption.

Claim 19 (Original): The method of claim 16 wherein said data is buffered for a length sufficient to complete a block cipher used to encrypt the data.

Claim 20 (Previously Presented): A method for secure communications between a client and one of a plurality of servers performed on an intermediary device, comprising:

establishing a communications session between the client and said one of said plurality of servers by receiving negotiation data from the client intended for the server and forwarding the negotiation data in modified form to the server, and receiving negotiation data from the server intended for the client and forwarding the negotiation data to the client to establish the client and the server as terminations for the communications session;

establishing a secure communications session between the client and the intermediary device;

maintaining a database of the secure communications session including information on the session/packet associations;

receiving encrypted application data from the client at the intermediary device by the secure communications session between the intermediary device and the client;

decrypting the application data; and

forwarding decrypted application data from the intermediary device to said one of said plurality of servers using the communications session established between the client and the server.

Claim 21 (Previously Presented): The method of claim 20 further including the steps of:

receiving at the intermediary device application data from the server destined for the client;

encrypting the application data at the intermediary device; and

forwarding the application data to the client along the secure communication session established between the intermediary device and the client.

Claim 22 (Original): The method of claim 20 wherein the method further includes the step of selecting one of the plurality of servers for each packet in the communications session and mapping all communications intended for the server to said one of said plurality of servers.

Claim 23 (Previously Presented): The method of claim 21 wherein forwarding the application to the data comprises receiving packets from said one of said plurality of servers and modifying the source and destination addresses of the packet to forward the packet to the client.

Claim 24 (Previously Presented): The method of claim 20, wherein said step of decrypting application data comprises decrypting data and forwarding said data on to said one of said plurality of servers via a secure network.

Claim 25 (Original): The method of claim 24 further including the step of receiving application data from said one of said plurality of servers, encrypting said data, and forwarding encrypted data to said client.

Claim 26 (Original): The method of claim 20 wherein said database includes an entry for a session comprising a session ID, a TCP Sequence number and an SSL session number.

Claim 27 (Original): The method of claim 20 wherein said entry further includes an initialization vector.

Claim 28 (Original): The method of claim 20 wherein said entry includes an expected ACK.

Claim 29 (Original): The method of claim 20 wherein said step of forwarding includes:

forwarding data which spans over multiple TCP segments and forwarding data which is not authenticated.

Claim 30 (Original): The method of claim 29 wherein said data is not buffered during decryption.

Claim 31 (Original): The method of claim 29 wherein said data is buffered for a length sufficient to complete a block cipher used to encrypt the data.

Claim 32 (Original): The method of claim 29 wherein said step of forwarding includes authenticating the decrypted data after a final segment of a multi-segment encrypted data stream is received.

Claim 33 (Previously Presented): An acceleration apparatus coupled to a public network and a secure network, communicating with a client via the public network and communicating with one of a plurality of servers via the secure network, comprising:

a network communications interface;

at least one processor;

programmable dynamic memory;

a communications channel coupling the processor, memory and network communications interface;

a client/server open communications session manager;

a client secure communication session manager;

a client/server secure communications session tracking database;

and

a data packet encryption and decryption engine,

wherein the acceleration apparatus is adapted to operate in a direct mode and a proxy mode,

wherein in the direct mode the acceleration apparatus decrypts data packets received from the client and forwards the decrypted data packets to one of the servers using a communication session negotiated by the client and the server,

wherein in the proxy mode the acceleration apparatus responds to the client on behalf of the server and forwards the decrypted data packets to the server using a communication session negotiated by the acceleration device and the server.

Claim 34 (Previously Presented): The apparatus of claim 33 wherein in the proxy mode the client open communications session manager and secure communication manager enable the apparatus as a TCP and SSL proxy for the server.

Claim 35 (Previously Presented): The apparatus of claim 33 wherein in the direct mode the communications session manager enables transparent secure and open communication between the client and the server.

Claim 36 (Cancelled).

Claim 37 (Previously Presented): The apparatus of claim 33 further including a load selection manager balancing the routing of multiple open and secure communications session between a plurality of clients and a plurality of servers based on current processing levels of the servers.

Claim 38 (Original): The apparatus of claim 33 wherein data packet encryption and decryption engine performs SSL encryption and decryption on data packets transmitted between the client and said at least one server.

Claim 39 (Original): The apparatus of claim 41 wherein the session tracking set maintains database having at least one record per communication session between the client and server.

Claim 40 (Original): The apparatus of claim 33 wherein said session tracking database includes a TCP sequence number and an SSL sequence number.

Claim 41 (Previously Presented): The apparatus of claim 33 further including a recovery manager coupled to the database.

Claim 42 (Original): The apparatus of claim 33 wherein said data is not buffered during decryption.

Claim 43 (Original): The apparatus of claim 33 wherein said data is buffered for a length sufficient to complete a block cipher used to encrypt the data.

Claim 44 (Previously Presented): The apparatus of claim 33 wherein said client/server open communications session manager performs an authentication process that discards at least a portion of the decrypted, unauthenticated packet application data from the client prior to receiving a final segment of the application data and authenticates the decrypted data using only the remaining portion of the application data.

Claim 45 (Previously Presented): A secure sockets layer processing acceleration device, comprising:

a communication engine establishing a secure communications session with a client device via an open network;

a server communication engine establishing an open communications session with a server via a secure network; and

an encryption and decryption engine operable on encrypted data packets received via the open communications session and on clear data received via the open communications session,

wherein the communication engine supports: (1) a direct mode in which decrypted data packets are forwarded to the servers using a communication session negotiated by the client and the server, and (2) a proxy mode in which the acceleration device responds to the client on behalf of the server and forwards the decrypted data packets to the server using the open communications session established by the acceleration device and the server.

Claim 46 (Previously Presented): The SSL acceleration device of claim 45 wherein when operating in direct mode the communication engine forwards modified communication session data to the server over the communication session between the client device and the server.

Claim 47 (Previously Presented): The SSL acceleration device of claim 45 wherein when operating in proxy mode the communication engine acts as a proxy for a plurality of servers in communication with the SSL acceleration device.

Claim 48 (Original): The SSL acceleration device of claim 45 further including a session tracking database interacting with the encryption and decryption engine tracking client and server communications.

Claim 49 (Original): The SSL acceleration device of claim 45 wherein the encryption and decryption engine includes a bufferless mode transmitting decrypted, unauthenticated data to a server.

Claim 50 (Previously Presented): The SSL acceleration device of claim 45 further including a load balancing engine that selects the server from a plurality of servers based on a load balancing algorithm that calculates current processing loads associated with each of the servers.

Claim 51 (Previously Presented): The method of claim 1, further comprising automatically switching the intermediate device from the direct mode to the proxy mode upon detecting a communication error associated with the direct mode.

Claim 52 (Previously Presented): The apparatus of claim 33, wherein the acceleration apparatus automatically switches from the direct mode to the proxy mode upon detection of a communication error associated with the communication session negotiated by the client and the server.

Claim 53 (Previously Presented): The SSL acceleration device of claim 45, wherein the communications engine automatically switches from the direct mode to the proxy mode upon detection of a communication error with the communication session negotiated by the client and the server.

APPENDIX: EVIDENCE

None

APPENDIX: RELATED PROCEEDINGS

None